

Bài 2: SỰ TƯƠNG ĐỒNG VÀ CÁC KHOẢNG CÁCH

I. Mục tiêu:

Sau khi thực hành xong, sinh viên nắm được:

- Khoảng cách giữa 2 điểm dữ liệu dùng chuẩn L_p với $p = 1, 2, \infty$.
- Độ đo thích ứng
- Độ đo tần suất xuất hiện ngược

II. Tóm tắt lý thuyết:

1. Chuẩn L_p :

Cho 2 điểm dữ liệu $\bar{X} = (x_1 \dots x_n)$ và $\bar{Y} = (y_1 \dots y_n)$, khoảng cách giữa 2 điểm dữ liệu này dùng chuẩn L_p được xác định như sau:

$$Dist(\bar{X}, \bar{Y}) = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{1/p}$$

Các trường hợp đặc biệt của chuẩn L_p là

- $p=1$ (Manhattan)

$$Dist(\bar{X}, \bar{Y}) = \left(\sum_{i=1}^n |x_i - y_i| \right)$$

- $p=2$ (Euclidean)

$$Dist(\bar{X}, \bar{Y}) = \left(\sum_{i=1}^n |x_i - y_i|^2 \right)^{1/2}$$

- $p = \infty$

$$Dist(\bar{X}, \bar{Y}) = \max_{1 \leq i \leq n} |x_i - y_i|$$

2. Độ đo thích ứng:

Xét 2 bản ghi $\bar{X} = (x_1 \dots x_d)$ và $\bar{Y} = (y_1 \dots y_d)$, sự tương đồng đơn giản nhất giữa 2 bản ghi này được xác định như sau

$$Sim(\bar{X}, \bar{Y}) = \sum_{i=1}^d S(x_i, y_i)$$

với $S(x_i, y_i)$ là sự tương đồng giữa các giá trị thuộc tính x_i, y_i . Lựa chọn đơn giản nhất cho $S(x_i, y_i)$ là

$$S(x_i, y_i) = \begin{cases} 1 & \text{nếu } x_i = y_i \\ 0 & \text{ngược lại} \end{cases}$$

3. Độ đo tần suất xuất hiện ngược:

Tần suất xuất hiện ngược là sự tổng quát hóa của độ đo thích ứng đơn giản. Độ đo này gắn thêm sự tương đồng giữa các thuộc tính thích ứng của 2 bản ghi bởi 1 hàm nghịch đảo của tần suất của giá trị thích ứng. Do đó, khi $x_i = y_i$ thì sự tương đồng $S(x_i, y_i)$ bằng với tần suất có trọng số nghịch đảo và ngược lại bằng 0. Cho $p_k(x)$ là một tỉ số của các bản ghi mà thuộc tính thứ k lấy giá trị x trong tập dữ liệu. Mặc khác,

$$S(x_i, y_i) = \begin{cases} \frac{1}{p_k(x)^2} & \text{nếu } x_i = y_i \\ 0 & \text{ngược lại} \end{cases}$$

III. Nội dung thực hành:

1. Download the Ionosphere data set from the UCI Machine Learning Repository



The screenshot shows a web browser window with the URL `archive.ics.uci.edu/ml/machine-learning-databases/ionosphere/`. The page title is "Index of /ml/machine-learning-databases/ionosphere". The page content includes a list of links: "Parent Directory", "Index", "ionosphere.data", and "ionosphere.names". At the bottom, there is a footer line: "Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips SVN/1.7.14 Phusion_Passenger/4.0.53 mod_perl/2.0.11 Perl/v5.16.3 Server at archive.ics.uci.edu Port 443".

- Đọc dữ liệu từ file:

```
>>> import pandas as pd
>>> import numpy as np
>>> url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/ionosphere/ionosphere.data'
>>> df = pd.read_csv(url)
>>> df
   .. ..
345 1 0 0.83508 0.08298 0.73739 ... 0.86660 -0.10714 0.90546 -0.04307 g
346 1 0 0.95113 0.00419 0.95183 ... 0.94066 -0.00035 0.91483 0.04712 g
347 1 0 0.94701 -0.00034 0.93207 ... 0.92459 0.00442 0.92697 -0.00577 g
348 1 0 0.90608 -0.01657 0.98122 ... 0.96022 -0.03757 0.87403 -0.16243 g
349 1 0 0.84710 0.13533 0.73638 ... 0.75747 -0.06678 0.85764 -0.06151 g
[350 rows x 35 columns]
```

- Xử lý dữ liệu (bỏ cột cuối):

```
>>> df = df[df.columns[:-1]]
>>> df
   1  0  0.99539 -0.05889 ...  0.42267 -0.54487  0.18641 -0.45300
0   1  0  1.00000 -0.18829 ... -0.16626 -0.06288 -0.13738 -0.02447
1   1  0  1.00000 -0.03365 ...  0.60436 -0.24180  0.56045 -0.38238
2   1  0  1.00000 -0.45161 ...  0.25682  1.00000 -0.32382  1.00000
3   1  0  1.00000 -0.02401 ... -0.05707 -0.59573 -0.04608 -0.65697
4   1  0  0.02337 -0.00592 ...  0.00000  0.00000 -0.00039  0.12011
... ..
345  1  0  0.83508  0.08298 ...  0.86660 -0.10714  0.90546 -0.04307
346  1  0  0.95113  0.00419 ...  0.94066 -0.00035  0.91483  0.04712
347  1  0  0.94701 -0.00034 ...  0.92459  0.00442  0.92697 -0.00577
348  1  0  0.90608 -0.01657 ...  0.96022 -0.03757  0.87403 -0.16243
349  1  0  0.84710  0.13533 ...  0.75747 -0.06678  0.85764 -0.06151

[350 rows x 34 columns]
```

- Khởi tạo các điểm point1, point2, point3 tương ứng là cột 0, 1, 2 của array và tính chuẩn $p=1, 2, \infty$

```
>>> array = df.values
>>> array
array([[ 1.      ,  0.      ,  1.      , ..., -0.06288, -0.13738, -0.02447],
       [ 1.      ,  0.      ,  1.      , ..., -0.2418 ,  0.56045, -0.38238],
       [ 1.      ,  0.      ,  1.      , ...,  1.      , -0.32382,  1.      ],
       ...,
       [ 1.      ,  0.      ,  0.94701, ...,  0.00442,  0.92697, -0.00577],
       [ 1.      ,  0.      ,  0.90608, ..., -0.03757,  0.87403, -0.16243],
       [ 1.      ,  0.      ,  0.8471 , ..., -0.06678,  0.85764, -0.06151]])
>>> point1 = array[:,0]
>>> point2 = array[:,1]
>>> point3 = array[:,2]
>>> dist1_2 = np.linalg.norm(point1 - point2)
>>> dist1_2
17.663521732655695
>>> dist1_3 = np.linalg.norm(point1 - point3)
>>> dist1_3
10.480973578165342
>>> #p=1
>>> dist01_2 = np.linalg.norm(point1 - point2, 1)
>>> dist01_3 = np.linalg.norm(point1 - point3, 1)
>>> dist01_2
312.0
>>> dist01_3
121.8844
>>> #p = infinity
>>> dist11_2 = np.linalg.norm(point1 - point2, np.inf)
>>> dist11_3 = np.linalg.norm(point1 - point3, np.inf)
>>> dist11_2
1.0
>>> dist11_3
1.0
```

2. Download the KDD Cup Network Intrusion Data Set for the UCI Machine Learning Repository.

← → ↻ archive.ics.uci.edu/ml/machine-learning-databases/kddcup99-mld/

Index of /ml/machine-learning-databases/kddcup99-mld

- [Parent Directory](#)
- [corrected.gz](#)
- [kddcup.data.gz](#)
- [kddcup.data_10_percent.gz](#)
- [kddcup.names](#)
- [kddcup.newtestdata_10_percent_unlabeled.gz](#)
- [kddcup.testdata.unlabeled.gz](#)
- [kddcup.testdata.unlabeled_10_percent.gz](#)
- [kddcup99.html](#)
- [task.html](#)
- [training_attack_types](#)
- [typo-correction.txt](#)

Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips SVN/1.7.14 Phusion_Passenger/4.0.53 mod_perl/2.0.11 Perl/v5.16.3 Server at archive.ics.uci.edu Port 443

- Khởi tạo tập dữ liệu chỉ chứa các thuộc tính cần thiết

```
>>> import pandas as pd
>>> import numpy as np
>>> df = pd.read_csv('D:\\Huynh\\DataMining\\data\\tuan2\\kddcup.data.csv')
>>> df
```

0	0	tcp	http	SF	215	45076	...	0.27	0.28	0.29	0.30	0.31	normal.
0	0	tcp	http	SF	162	4528	...	0.0	0.00	0.00	0.00	0.00	normal.
1	0	tcp	http	SF	236	1228	...	0.0	0.00	0.00	0.00	0.00	normal.
2	0	tcp	http	SF	233	2032	...	0.0	0.00	0.00	0.00	0.00	normal.
3	0	tcp	http	SF	239	486	...	0.0	0.00	0.00	0.00	0.00	normal.
4	0	tcp	http	SF	238	1282	...	0.0	0.00	0.00	0.00	0.00	normal.
...
1048570	0	tcp	http	SF	318	2681	...	0.0	0.01	0.01	0.02	0.02	normal.
1048571	0	tcp	http	SF	316	2539	...	0.0	0.01	0.01	0.02	0.02	normal.
1048572	0	tcp	http	SF	320	9693	...	0.0	0.01	0.01	0.02	0.02	normal.
1048573	0	tcp	http	SF	317	2186	...	0.0	0.01	0.01	0.02	0.02	normal.
1048574	0	tcp	http	SF	315	2284	...	0.0	0.01	0.01	0.02	0.02	normal.

```
[1048575 rows x 42 columns]
>>> #xoa cot 2, 3, 4 va cot cuoi cung
>>> df= df.drop(['tcp', 'http', 'SF', 'normal.'], axis=1)
>>> df
```

0	215	45076	0.1	0.2	0.3	...	0.26	0.27	0.28	0.29	0.30	0.31
0	0	162	4528	0	0	0	...	1.00	0.0	0.00	0.00	0.00
1	0	236	1228	0	0	0	...	0.50	0.0	0.00	0.00	0.00
2	0	233	2032	0	0	0	...	0.33	0.0	0.00	0.00	0.00
3	0	239	486	0	0	0	...	0.25	0.0	0.00	0.00	0.00
4	0	238	1282	0	0	0	...	0.20	0.0	0.00	0.00	0.00
...
1048570	0	318	2681	0	0	0	...	0.00	0.0	0.01	0.01	0.02
1048571	0	316	2539	0	0	0	...	0.00	0.0	0.01	0.01	0.02
1048572	0	320	9693	0	0	0	...	0.00	0.0	0.01	0.01	0.02
1048573	0	317	2186	0	0	0	...	0.00	0.0	0.01	0.01	0.02
1048574	0	315	2284	0	0	0	...	0.00	0.0	0.01	0.01	0.02

```
[1048575 rows x 38 columns]
```

- Tính các láng giềng gần nhất cho mỗi điểm dữ liệu sử dụng
 - Độ đo thích ứng (match measure)
 - Độ đo tần suất xuất hiện ngược (inverse occurrence frequency measure)

3. Yêu cầu:

- Tính các chuẩn $p = 1, 2, \infty$ cho các cột còn lại của array trong bài 1.
- Tính các láng giềng gần nhất cho mỗi điểm dữ liệu ở bài 2.